Automatic visual inspection of printed circuit board for defect detection and classification

Conference Paper · March 2017

DOI: 10.1109/WiSPNET.2017.8299858

3 authors, including:

Ishan Dave
University of Central Florida
11 PUBLICATIONS **47** CITATIONS
SEE PROFILE

Kishor Upla
Sardar Vallabhbhai National Institute of Technology
53 PUBLICATIONS **147** CITATIONS
SEE PROFILE

Automatic Visual Inspection of Printed Circuit Board for Defect Detection and Classification

Vikas Chaudhary, Ishan R. Dave and Kishor P. Upla
S. V. National Institute of Technology, Surat, India
{vrnvikas1994, ishandave95, kishorupla}@gmail.com

***Abstract*— Inspection of printed circuit board (PCB) has been a crucial process in the electronic manufacturing industry to guarantee product quality & reliability, cut manufacturing cost and to increase production. The PCB inspection involves detection of defects in the PCB and classification of those defects in order to identify the roots of defects. In this paper, all 14 types of defects are detected and are classified in all possible classes using referential inspection approach. The proposed algorithm is mainly divided into five stages: Image registration, Pre-processing, Image segmentation, Defect detection and Defect classification. The algorithm is able to perform inspection even when captured test image is rotated, scaled and translated with respect to template image which makes the algorithm rotation, scale and translation in-variant. The novelty of the algorithm lies in its robustness to analyze a defect in its different possible appearance and severity. In addition to this, algorithm takes only 2.528 seconds to inspect a PCB image. The efficacy of the proposed algorithm is verified by conducting experiments on the different PCB images and it shows that the proposed afgorithm is suitable for automatic visual inspection of PCBs.**

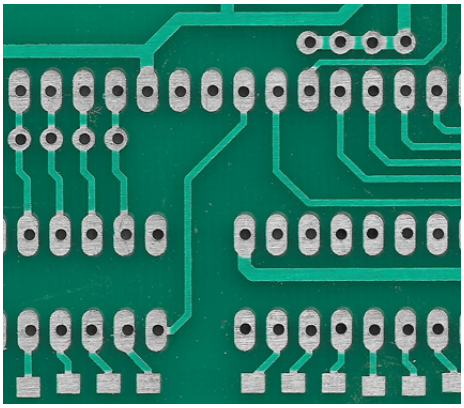
***Index Terms*— Printed Circuit Boards, Automatic Visual Inspection, Machine Vision**

I. Introduction

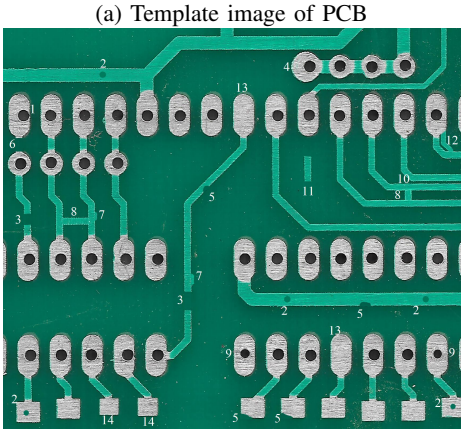
Production of PCB is an essential component in the electronics industries. The performance of a PCB is significantly dependent on its quality and reliability. A defective PCB may result in undesirable circuit behaviour and may end up in a defective product. Due to this PCB inspection is a crucial process in electronics industries. The aim of this inspection process is to assure 100% quality of all parts, which costs the most in manufacturing [1], [2]. Conventionally, human operators are involved in the visual inspection of PCB to detect and classify the defects. This conventional manual inspection process is time-consuming, tedious and error-prone. Also, the results of inspection may vary person to person due to human inconsistency. The quality control problem can be solved by using developments in computer vision field. In order to make PCB inspection process fast and reliable, automatic visual inspection (AVI) systems is more useful in industries.

AVI approaches are mainly divided into three different methods: referential, non-referential and hybrid methods [3]. In the referential method, the given test image of the PCB is compared with its template image in order to locate defects. The non-referential method is the design rule based method which verifies whether the design of PCB is in predefined limits or not. But the disadvantage of the non-referential method is that it is not able to identify defects in their distorted appearance. The hybrid method is the combination of both referential and non-referential methods. But, the disadvantage of the hybrid method is its higher computational complexity.

The template and defective images of PCB are shown in Fig. 1(a) and (b), respectively. There are 14 types of known defects in PCB as shown in Fig. 1(b).



(a) Template image of PCB



(b) Test image of PCB with defects: (1) Breakout, (2) Pinhole, (3) Open circuit, (4) Under etch, (5)Mouse bite, (6) Missing conductor, (7) Spur, (8) Short, (9) Wrong size hole, (10) Conductor too close, (11) Spurious Copper, (12) Excessive short, (13) Missing hole and (14) Over etch.

Fig. 1: PCB images for referential method

In the literature, many authors have attempted to detect and classify the defects in PCB image using various methods. Wu

et al. [4] use the referential method in order to detect and classify the defects into seven defined groups. The classification is performed using three indices of a defect based on type and number of objects. Putera et al. [5] utilise the area property of defect in order to classify it into seven defined groups, with maximum four defects in a group. Similarly, Nakagawa et al. [6] propose a referential method and it classifies the defects into three defined classes. The method proposed in [6] classifies the PCV image using multiple support vector machine (SVM) which is trained using 24 various features of defect candidate. In [7], authors propose a referential method by using the edge grey gradient of the PCB image in order to classify defects into 5 defined classes. Furthermore, Kumar et al. [8] propose a non-referential method to classify defects into 4 defined classes. However, the limitation of this method is that it can classify only one defect per image. The classification of defects in their desired class is as important as detection of defects. This classification is an essential process in order to identify the roots of defects. Ap per the recent record no author has tried to classify all 14 PCB defects into all 14 possible classes.

In this paper, we propose a referential method to detect and classify the defects of PCB into all possible 14 classes. The proposed algorithm is mainly divided into five stages: Image registration, Pre-processing, Image segmentation, Defect detection and Defect classification. Firstly, in Section II, image registration technique is elaborated in order to remove variation in captured test image like rotation, scale and translation with respect to template image of the same PCB. Next to that in Section III, pre-processing steps are explained in order to reduce noise and enhance the image details. In Section IV, the image segmentation is explained. The defect detection and classification are the topics of discussion in Section V and VI, respectively. Results and timing report of the algorithm is shown in Section VII. Finally, conclusion is drawn in Section VIII. The complete block schematic of the proposed algorithm is depicted in Fig. 2.

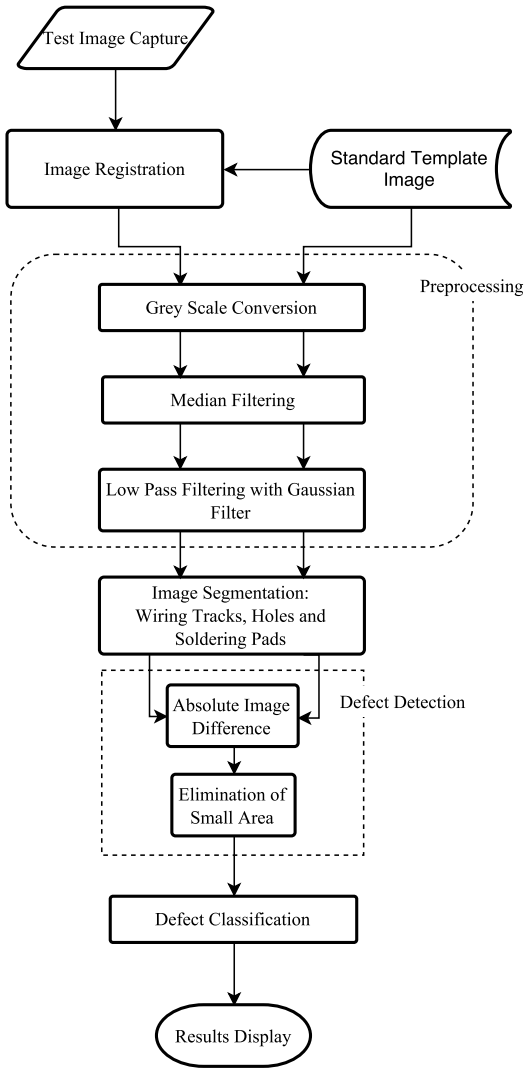


Fig. 2: Block schematic of the proposed algorithm

II. Image Registration

Test PCB is scanned by HP Laserjet scanner in order to get the test PCB image. This image may have variations in terms of rotation and translation with respect to the template image as shown in Fig. 3. Such variations can be removed by using image registration techniques [9]. Block diagram of image registration is shown in Fig. 4. The test image and template images are converted into grey scale by using Equation (1)

$$Greylevel = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B, \quad (1)$$

where R, G and B are the red, green and blue channels in am color image. Next to this is to extract the features from the both template and test images. Since this process is most time-consuming in image registration algorithm it is desirable to use fast algorithm for this. Table I shows the time to execute registration process using different feature extraction techniques. Features from accelerated segment test (FAST) algorithm [10] is used since it takes lesser time to execute than other techniques as mentioned in Table I.

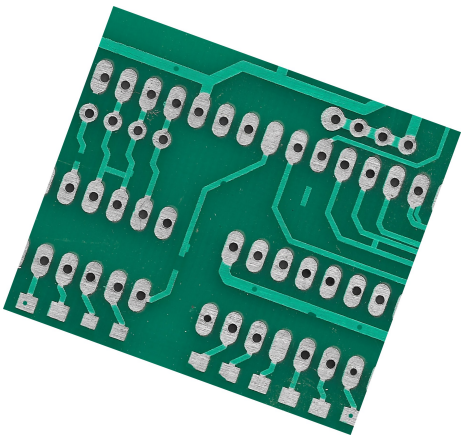


Fig. 3: Un-registered test PCB image

The extracted features are matched using sum of squared difference (SSD) metric. Geometric transformation matrix is then estimated from matched features using m-estimator sample consensus (MSAC) algorithm [16]. The estimated

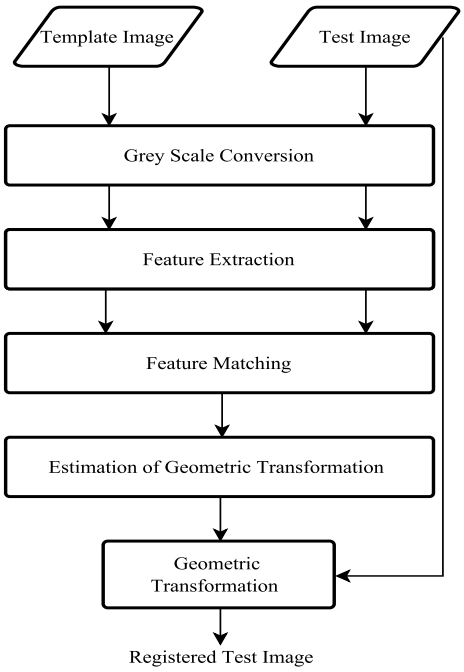


Fig. 4: Block diagram of image registration process

TABLE I: Registration time using different feature extraction methods

Feature extraction method	Execution time (second)
SURF [11]	2.04
Harris [12]	2.635
BRISK [13]	1.497
FAST [10]	1.143
MSER [14]	4.411
MinEigen [15]	5.2

transformation is then applied to test image in order to get registered image. The output of image registration is shown in Fig. 5.

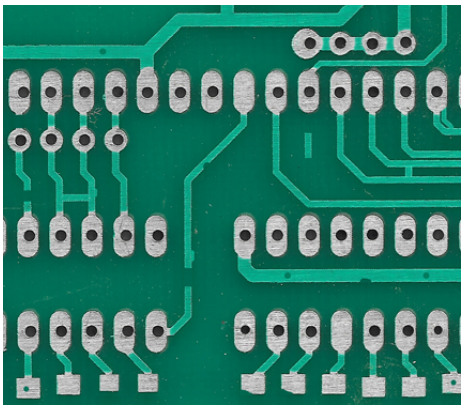
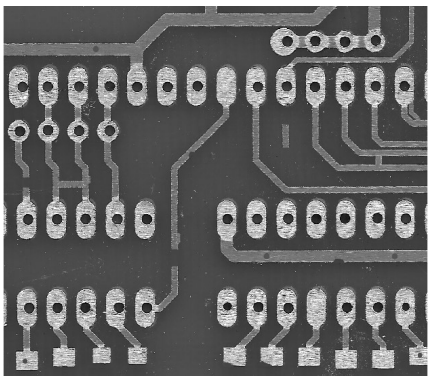


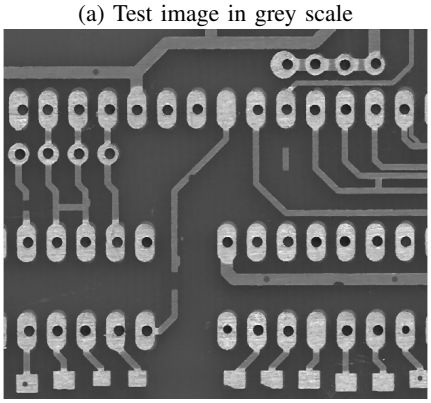
Fig. 5: Output of image registration process

III. Pre-processing

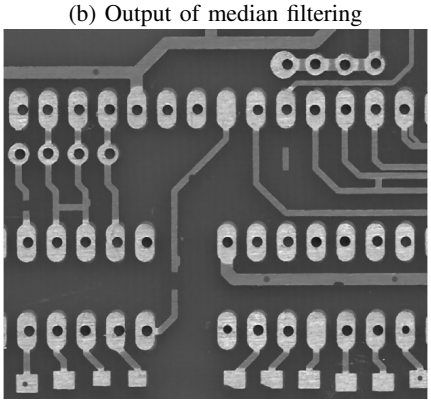
The acquired PCB images may consist of noise such as salt and pepper noise. Also, these images may have high variations in intensity levels due to different lighting conditions, which leads to improper binarization of image. The purpose of pre-processing is to remove noise and enhance the image details. Fig. 6(a) shows the grey scale image of PCB using Equation (1). Median filter of mask size 7×7 is then applied to the grey scale image in order to remove salt and pepper noise. The output image is depicted in Fig.6(b). Next to the denoising, high-intensity variation is suppressed by applying Gaussian low-pass filtering (standard deviation=1). In Fig. 6(c) we have displayed a Gaussian low-pass filtered image.



(a) Test image in grey scale



(b) Output of median filtering



(c) Output of low-pass filtering

Fig. 6: Preprocessing steps

IV. Image Segmentation

Next to pre-processing step, is the step of image segmentation. The purpose of image segmentation is to represent the image in different parts (sets of pixels), which makes the representation of image more meaningful. In PCB image,

there are mainly three interested parts: (1) wiring tracks (2) soldering pads and (3) holes. In the proposed method, we use histogram thresholding method, followed by mathematical morphology operations to segment the PCB image into mentioned parts. Fig. 7 shows the normalised histogram of the PCB image. Wiring tracks and soldering pads are

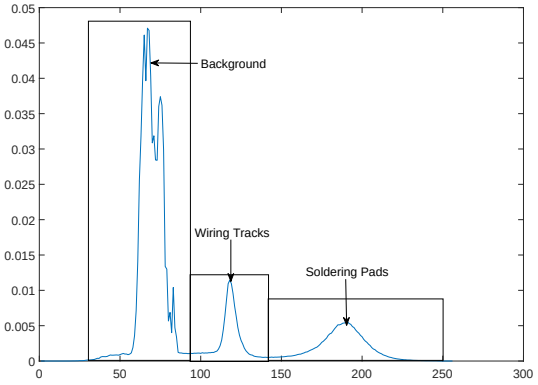


Fig. 7: Normalized histogram of PCB image

obtained by using upper and lower threshold points as shown in Equation (2) and (3), respectively.

$$Wiring\,tracks = \begin{cases} 1, \, if\, 95 < greylevel < 140; \\ 0, \, else \end{cases} \quad (2)$$

$$Soldering\,pads = \begin{cases} 1, \, if\, greylevel > 140; \\ 0, \, else \end{cases} \quad (3)$$

The zero regions inside the soldering pads represent holes. These zero regions are filled with region filling operations. Soldering pads regions are subtracted from this filled image in order to get the regions of holes. The segmented images are shown in Fig. 8.

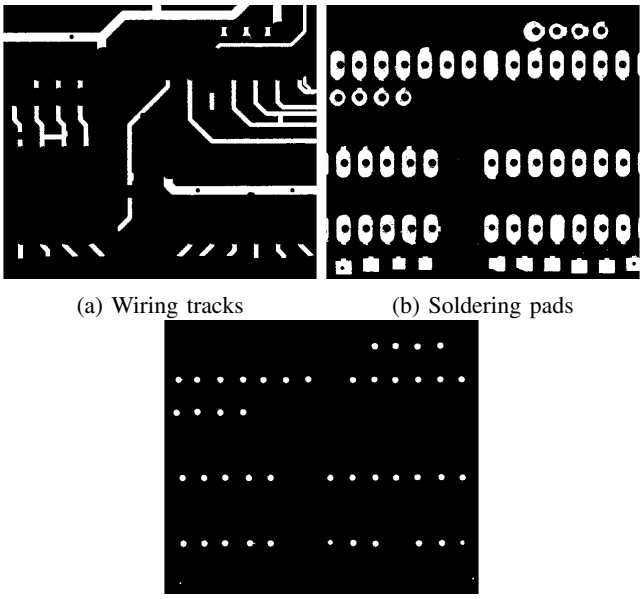
V. Defect Detection

The segmented images (wiring tracks, soldering pads and holes) of test and template images differ from each other due to defects in testing PCB image. So, the defects can be simply detected by image subtraction. These defects are of two types: (1) positive defects (PD) and (2) negative defects (ND). As shown in Equation (4) positive defects can be detected by subtracting segmented template images from the corresponding segmented testing images; and vice versa for negative defects (Equation (5)).

$$PD_i = testing_i - template_i \quad (4)$$

$$ND_i = template_i - testing_i, \quad (5)$$

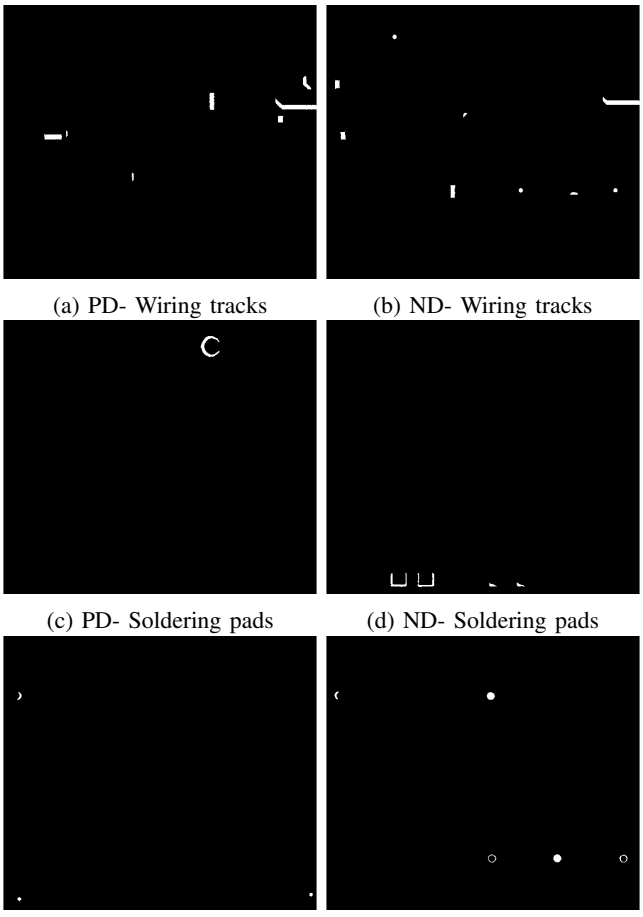
where, i=wiring, tracks, soldering, pads and holes. Uneven binarization of edges also causes small differences between test and template images. This kind of small differences can be eliminated by area filtering. Detected defects after area filtering are shown in Fig. 9.



(a) Wiring tracks (b) Soldering pads

(c) Holes

Fig. 8: Segmented images



(a) PD- Wiring tracks (b) ND- Wiring tracks

(c) PD- Soldering pads (d) ND- Soldering pads

(e) PD- Holes (f) ND- Holes

Fig. 9: Defect detection

VI. Defect Classification

A. Defects related to wiring tracks

Positive and negative defects related to wiring tracks are shown in Table II. Centroid and maximum radius of

TABLE II: Defects related to wiring tracks

Positive Defects (PDW)	Spur, Short, Spurious copper, Excessive short, Conductor too close
Negative Defects (NDW)	Pinhole, Mouse bite, Open circuit, Missing conductor, Conductor too close

defects are found from PDW and NDW images by using 8-connected components. To check the neighborhood of a defect, a square region (length= maximum radius of defect, centroid= centroid of defect) is cropped from the segmented wiring track image of template image (WT). The flowchart of defect classification is shown in Fig. 10 and 11 for positive and negative defects, respectively. Here, WT and SP represents wiring track segmented image and soldering pads segemented image, respectively for template image. WT1 represents segmented wiring track image of testing image (Fig. 8)(a).

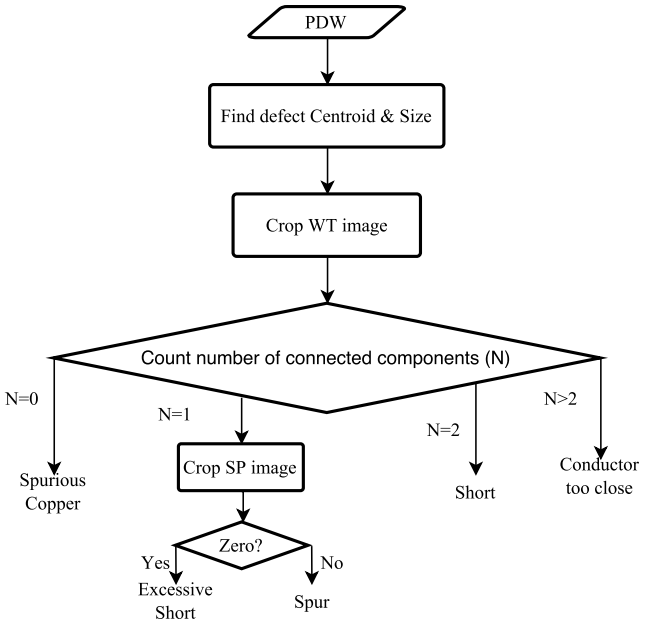


Fig. 10: Classification of wiring track defects (positive)

B. Defects related to soldering pads

Positive and negative defects related to soldering pads are shown in Table III. Under and Over etch defects have larger

TABLE III: Defects related to soldering pads

Positive Defects (PDS)	Underetch, Spur
Negative Defects (NDS)	Overetch, Mousebite

area (≈ 2000) than the area of spur and mouse bite defects (≈ 400). Using this difference in area soldering pad defects are classified as shown in Fig. 12.

C. Defects related to holes

Positive and negative defects related to holes are shown in Table IV. Bold fonts in Table IV represents shape of the defect. There are mainly three shapes observed in hole

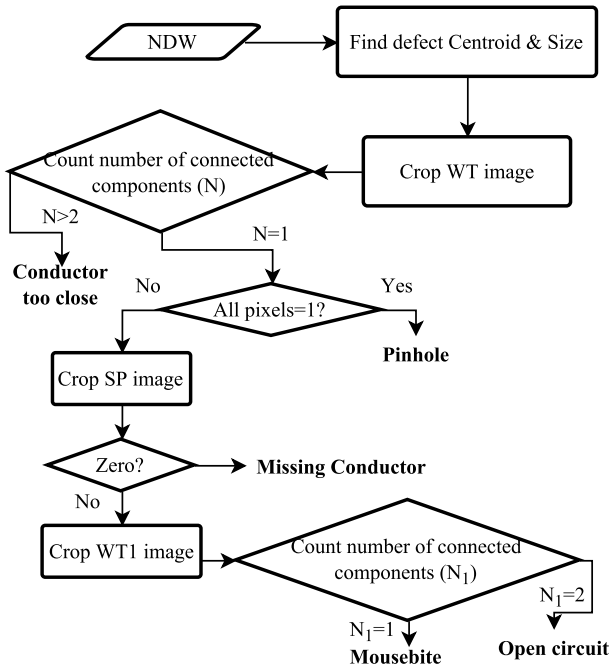


Fig. 11: Classification of wiring track defects (negative)

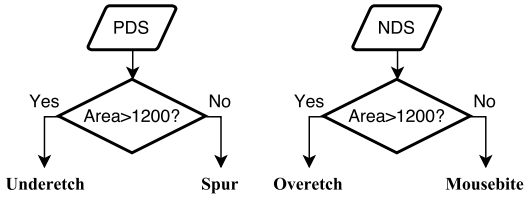


Fig. 12: Classification of soldering pad defects

TABLE IV: Defects related to holes

Positive Defects (PDH)	Pinholes (**Circle**), Wrong size (Big) hole (**Ring**) and Breakout (**Half-moon**)
Negative Defects (NDH)	Missing holes (**Circle**), Wrong size (Small) hole (**Ring**) and Breakout (**Half-moon**)

defects: (1) circle (2) ring and (3) half-moon as shown in Fig. 13. To make the classification process invariant to rotation and scale, Hu's 2^{nd} invariant moment [17] is used to classify these shapes. Hu's 2nd moment for circle, ring and half-moon shapes are 3×10^{-5}, 40×10^{-5} and 6390×10^{-5}, respectively.

(a) Circle shaped defect (b) Ring shaped defect (c) Half-moon shaped defect

Fig. 13: Hole defects shapes

VII. Results

The final result obtained after classification step is shown in Fig. 14. One can see that all the defects are successfully detected and classified into correct classes. In addition to this, the proposed algorithm takes 2.528 seconds to execute the inspection of a PCB image. The complete timing data for each step of algorithm is depicted in Table V. In the proposed method, except soldering pad defects, the proposed algorithm uses scale invariant parameters (e.g. number of connected component and shape based moment of defect) instead of using scale based parameters like area of defect. Scale invariant features make the classification process robust to defect severity.

TABLE V: Timimg report of the proposed algorithm

Step	Time (second)
Registration	1.143
Preprocessing	0.223
Defect Detection	0.001
Defect Classification	1.161
Total	**2.528**

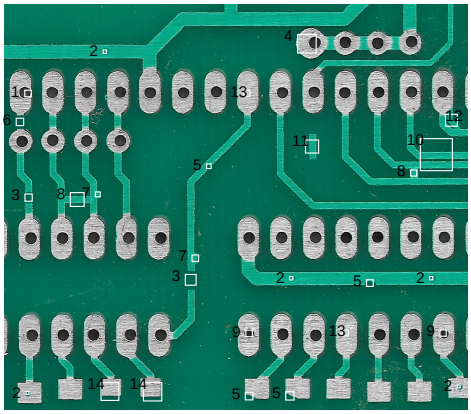


Fig. 14: Result generated by the proposed algorithm

VIII. Conclusion

In this paper, we have proposed a novel method to detect and classify all 14 types of defects of PCB using referential inspection method. Novelty of the algorithm is that it classifies all type of defects which is robust to defect appearance and severity. The testing (defective) image is aligned with the template (standard) image using image registration techniques. Noise in the image is reduced using median filtering. Furthermore, Gaussian low-pass filtering is used in order to avoid uneven binarization due to sharp transitions at edges. The PCB image is segmented in three parts: wiring tracks, soldering pads and holes in order to analyze defects in different parts of PCB image. The defect is detected using two-step process: image subtraction followed by area filtering to eliminate small areas after subtraction. After detecting defects, each defect is classified using various region properties like number of connected components, shape based descriptors and area.

The proposed algorithm is able to identify all 14 types of PCB defects, which is not covered in the state-of-the-art algorithms. Also, proposed method takes only 2.528 second to inspect a PCB image which makes it more suitable for AVI. The algorithm is useful in electronics manufacturing industries to inspect PCB quickly and accurately, that may lead to reduced production time and improvement in overall quality and reliability of product.

Acknowledgement

Authors would like to acknowledge the help supported by:

1. Mr. Jagdish Namera, Circuitronix, Gandhinagar
2. Mr. Amit Patel, Circuitronix, Gandhinagar
3. Mr. Archit Bhavsar, B.E., DCEIT, SVBIT

References

[1] R. T. Chin and C. A. Harlow, "Automated visual inspection: A survey," *IEEE transactions on pattern analysis and machine intelligence*, no. 6, pp. 557–573, 1982.

[2] R. T. Chin, "Automated visual inspection: 1981 to 1987," *Computer Vision, Graphics, and Image Processing*, vol. 41, no. 3, pp. 346–381, 1988.

[3] M. Moganti, F. Ercal, C. H. Dagli, and S. Tsunekawa, "Automatic pcb inspection algorithms: a survey," *Computer vision and image understanding*, vol. 63, no. 2, pp. 287–313, 1996.

[4] W.-Y. Wu, M.-J. J. Wang, and C.-M. Liu, "Automated inspection of printed circuit boards through machine vision," *Computers in industry*, vol. 28, no. 2, pp. 103–111, 1996.

[5] S. H. I. Putera, S. F. Dzafaruddin, and M. Mohamad, "Matlab based defect detection and classification of printed circuit board," in *Digital Information and Communication Technology and it's Applications (DICTAP), 2012 Second International Conference on*. IEEE, 2012, pp. 115–119.

[6] T. Nakagawa, Y. Iwahori, and M. Bhuyan, "Defect classification of electronic board using multiple classifiers and grid search of svm parameters," in *Computer and Information Science*. Springer, 2013, pp. 115–127.

[7] S. Ren, L. Lu, L. Zhao, and H. Duan, "Circuit board defect detection based on image processing," in *Image and Signal Processing (CISP), 2015 8th International Congress on*. IEEE, 2015, pp. 899–903.

[8] S. Kumar, Y. Iwahori, and M. Bhuyan, "Pcb defect classification using logical combination of segmented copper and non-copper part," in *Proceedings of International Conference on Computer Vision and Image Processing*. Springer, 2017, pp. 523–532.

[9] R. C. Gonzalez and R. E. Woods, *Digital Image Processing (3rd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2006.

[10] E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking," in *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, vol. 2. IEEE, 2005, pp. 1508–1515.

[11] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European conference on computer vision*. Springer, 2006, pp. 404–417.

[12] C. Harris and M. Stephens, "A combined corner and edge detector." in *Alvey vision conference*, vol. 15, no. 50. Citeseer, 1988, pp. 10–5244.

[13] S. Leutenegger, M. Chli, and R. Y. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2548–2555.

[14] D. Nistér and H. Stewénius, "Linear time maximally stable extremal regions," in *European Conference on Computer Vision*. Springer, 2008, pp. 183–196.

[15] J. Shi *et al.*, "Good features to track," in *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on*. IEEE, 1994, pp. 593–600.

[16] P. H. Torr and A. Zisserman, "Mlesac: A new robust estimator with application to estimating image geometry," *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 138–156, 2000.

[17] M.-K. Hu, "Visual pattern recognition by moment invariants," *IRE transactions on information theory*, vol. 8, no. 2, pp. 179–187, 1962.